UNITED STATES PATENT APPLICATION

for

# METHOD AND APPARATUS FOR SPEECH RECOGNITION USING LATENT SEMANTIC ADAPTATION

Inventor:

Jerome R. Bellegarda

Prepared by:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, L.L.P.
12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025
(503) 684-6200

"Express Mail" mailing label number: <u>EL034435545US</u>
Date of Deposit: _____
I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post Office to Addressee" service on the date indicated above and that this paper or fee has been addressed to the Assistant Commissioner for Patents, Washington, D. C. 20231

_____
(Typed or printed name of person mailing paper or fee)

_____
(Signature of person mailing paper or fee)       (Date signed)

# METHOD AND APPARATUS FOR SPEECH RECOGNITION USING LATENT SEMANTIC ADAPTATION

5
## BACKGROUND OF THE INVENTION

### Field of the Invention

The present invention relates generally to pattern recognition. More particularly, this invention relates to speech recognition systems using latent semantic analysis.

### Copyright Notice/Permission

### Background

As computer systems have evolved, the desire to use such systems for pattern recognition has grown. Typically, the goal of pattern recognition systems is to quickly

20 provide accurate recognition of input patterns. One type of pattern recognition system is a voice recognition system, which attempts to accurately identify a user's speech. Another type of pattern recognition is a handwriting recognition system. A speech recognizer discriminates among acoustically similar segments of speech to recognize words, while a handwriting recognizer discriminates among strokes of a pen to recognize

25 words.

An important advancement in speech recognition technology is the use of semantic pattern recognition known as semantic language modeling. Semantic language

modeling uses the context of the spoken words to decide which words are most likely to appear next, the context referring to the domain or subject matter of the words as well as the style. For example, a speech recognition application using semantic language modeling will favor the word sequence "recognize speech" over "wreck a nice beach"

5     when the subject matter is speech processing, and vice versa when the subject matter has to do with vacations at the beach.

In semantic language modeling, the domain and style of the spoken words is captured using latent semantic analysis (LSA). LSA is a modification of a paradigm that was first formulated in the context of information retrieval and reveals meaningful

10    associations in language based on semantic patterns previously observed in a corpus of language representative of a particular domain and style, for example, a training corpus having to do with speech processing vs. vacations at the beach. The semantic patterns are word-document co-occurrences that appear in the training corpus, where the corpus is comprised of a collection of one or more documents that contain paragraphs and

15    sentences or other collections of words representative of the domain and style.

The semantic knowledge represented by the semantic patterns is encapsulated in a continuous vector space, referred to as the LSA space, by mapping those word-document co-occurrences into corresponding word and document vectors that characterize the position of the words and documents in the LSA space. During speech recognition, any

20    new words or documents are first mapped onto a point in the LSA space, and then compared to the existing word and document vectors in the space using a similarity measure, a process referred to as semantic inference. Those new words and documents that map most closely to the existing word and document vectors in the LSA space are recognized over those that do not.

25    A limitation in current implementations of speech recognition applications using semantic language modeling is that the LSA space is a fixed semantic space. This means that semantic patterns not observed in the training corpus cannot be captured and later

exploited during speech recognition. As a result, changes in the domain of the speech, or even just changes in the style of the speech, may not be properly recognized. In the case of financial news, for example, this means that an LSA-based speech recognition application trained on a collection of documents, say, from the Wall Street Journal, will not perform optimally on new documents from the Associated Press, and vice versa. The use of a fixed semantic space is particularly deleterious in applications with many heterogeneous domains, such as an information retrieval system, since no database is big enough to contain a training corpus representative of all domains. It is also less than ideal for horizontal (i.e. non-specialized) dictation applications, because the same user typically adopts different styles in different contexts, for example the formal style of a business letter vs. the informal style of a personal letter.

Distributed training seeks to overcome some of the limitations of a fixed semantic space by creating a distinct semantic space for each usage condition. Thus, using the financial news example, there would be one LSA space for the Wall Street Journal, and another LSA space for the Associated Press. However, it is often impossible to predict ahead of time which kind(s) of text the end user will want to process, and even when that can be done, for most narrowly defined contexts and styles it may be challenging to gather enough data to reliably train the speech recognition system.

Explicit modeling also seeks to overcome some of the limitations of a fixed semantic space by including a task (i.e. domain) and/or style component into the LSA paradigm. For example, it has been suggested to define a stochastic matrix to account for the way style modifies the frequency of words (C.H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala, *"Latent Semantic Indexing: A Probabilistic Analysis,"* in *Proc. 17th ACM Symp. Princip. Database Syst.*, Seattle, WA, 1998). However, this approach makes the assumption -- largely invalid -- that the influence of style on word frequency is independent of the underlying domain.

Another approach to the problem of a fixed semantic space is to re-compute the LSA space to account for the new words and documents as they become available. One way is simply to re-compute the LSA space from scratch, referred to as full re-computation. Another way is to re-compute the LSA space from scratch, but keeping the dimension of the LSA space constant, referred to as constant dimension re-computation. But full or constant dimension re-computation requires significant additional processing. The additional processing is undesirable since it consumes additional central processor unit (CPU) cycles and degrades responsiveness.

Yet another approach to the problem of a fixed semantic space is to adapt the LSA space to account for the new documents and new words in the new documents as they become available by using traditional "folding-in" to incorporate new variants in the existing LSA space, referred to as baseline adaptation. While less computationally intensive, baseline adaptation results in speech misclassification error rates of unacceptably high levels. What is needed, therefore, is an improved method and apparatus for using semantic language modeling in a speech recognition system to more accurately recognize speech.

## SUMMARY OF THE INVENTION

A method and apparatus for speech recognition using latent semantic adaptation is described herein. According to one aspect of the present invention, a method for recognizing speech comprises generating a latent semantic analysis (LSA) space for a

5 collection of documents and the words appearing in those documents, and to continually adapt the LSA space with new documents as they become available. Adaptation of the LSA space is optimally two-sided, taking into account the new documents, as well as the new words that appear in those new documents. Alternatively, adaptation is one-sided, taking into account the new documents, but discarding any new words appearing in those

10 new documents.

According to one aspect of the present invention, a computer-readable medium has executable instructions to cause a computer to perform a method to generate a speech recognition database comprising generating an LSA space for a collection of documents and the words appearing in those documents, and continually adapting the LSA space

15 with new documents or both new documents and new words, as they become available.

According to one aspect of the present invention, an apparatus for recognizing speech includes an adapted LSA space generator. The adapted LSA space generator generates an LSA space from a collection of documents and the words appearing in those documents, and continually adapts the LSA space with new documents or both new

20 documents and new words, as they become available.

# BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings, in which like references indicate similar elements and in which:

5    FIG. 1 is a block diagram that illustrates the use of latent semantic adaptation in the context of a speech recognition system using semantic inference, in accordance with one embodiment of the present invention;

FIG.2 is a block diagram overview of some of the components of latent semantic analysis (LSA), in accordance with one embodiment of the present invention;

10    FIG. 3 is an overview of selected components of the basic LSA paradigm, in accordance with one embodiment of the present invention;

FIG. 4 is an overview of selected components of the adaptive LSA paradigm, in accordance with one embodiment of the present invention;

FIG. 5 is an overview of selected components of the matrix transformation of the adaptive LSA paradigm, in accordance with one embodiment of the present invention;

15    FIG. 6 is an overview of selected components of the vector transformation of the adaptive LSA paradigm, in accordance with one embodiment of the present invention;

FIG. 7 is an overview of selected components of prior art baseline adaptation;

FIG. 8 is a flowchart illustrating the process followed in generating and adapting an LSA space, in accordance with one embodiment of the present invention;

20    FIG. 9 illustrates one embodiment of a computing device suitable for use with one embodiment the present invention; and

FIG. 10 illustrates a graph comparing the misclassification error rates of an embodiment of the present invention as compared to other methods of re-computation

25    and adaptation of the LSA space.

# DETAILED DESCRIPTION

In the following description, various aspects of the present invention will be described. However, it will be understood by those skilled in the art that the present invention may be practiced with only some or all aspects of the present invention. For purposes of explanation, specific numbers, materials and configurations are set forth in order to provide a thorough understanding of the present invention. However, it will also be apparent to those skilled in the art that the present invention may be practiced without these specific details.

Parts of the description will be presented in terms of operations performed by a computer system, using terms such as data, flags, bits, values, characters, strings, numbers and the like, consistent with the manner commonly employed by those skilled in the art to convey the substance of their work to others skilled in the art. As is well understood by those skilled in the art, these quantities take the form of electrical, magnetic, or optical signals capable of being stored, transferred, combined, and otherwise manipulated through mechanical and electrical components of the computer system; and the term computer system includes general purpose as well as special purpose data processing machines, systems, and the like, that are standalone, adjunct or embedded.

Additionally, various operations will be described as multiple discrete steps in turn in a manner that is helpful in understanding the present invention. However, the order of description should not be construed as to imply that these operations are necessarily order dependent, in particular, the order of their presentations.

The present invention provides a method and apparatus for speech recognition using semantic language modeling. Specifically, the method and apparatus use latent semantic adaptation to reduce the misclassification error rate in speech recognition applications without sacrificing computational efficiency.

Latent semantic adaptation is a process of using latent semantic analysis (LSA) to capture the semantic patterns appearing in a training corpus of language by mapping them

into a continuous vector space, referred to as an LSA space, and continually adapting the LSA space with new semantic patterns as they appear over time. The training corpus is a collection of documents, where documents are instances of sentences, phrases, or other word groupings representative of a particular domain and style of language composition.

5        Because semantic patterns that are not present in the training corpus cannot be captured, current implementations of semantic language modeling using LSA exhibit a relatively high sensitivity to changes in both the domain and style of language composition when using a particular speech recognition application. For example, as noted previously in the financial news example, an LSA-based speech recognition

10      application trained on a collection of documents, say, from the Wall Street Journal, will not perform optimally on new documents from the Associated Press, and vice versa.

In accordance with the method and apparatus of the present invention, however, the incremental adaptation of the LSA space improves performance by continually modifying the LSA space on the basis of new words and documents as they become

15      available, where the new words and documents represent changes in either the domain or style of composition. As a result, any change in domain and/or style gradually gets reflected in the evolution of the LSA space, so that even new documents that do not closely conform to the training corpus (e.g. documents that contain several new words) can still be successfully processed.

20      Among other advantages, latent semantic adaptation implemented in accordance with the method of the present invention accommodates new documents of virtually any size and number, is capable of taking advantage of out-of-vocabulary words present in the new documents, and is computationally efficient since it does not require re-computation of the LSA space or multiple matrix inversions. The latent semantic adaptation of the

25      present invention is applicable to all applications of LSA, including semantic inference, dictation, information retrieval, and word and document clustering.

FIG. 1 is a block diagram that illustrates the use of latent semantic adaptation in accordance with one embodiment of the present invention, in the context of an LSA application using semantic inference in a speech recognition system 100. A speech recognition unit 104 receives an audio input 102 and, using acoustic models 106 and a

5     language model 108, generates a sequence of words and documents 110. The audio input 102 is audio data that is input to the speech recognition system 100 and is intended to represent any type of audio data. Typically, the audio input 102 is a digitized representation of a human voice.

According to one embodiment of the present invention, the acoustic models 106

10     are hidden Markov models. Alternate embodiments can use different types of acoustic models, and any of a variety of conventional acoustic models other than hidden Markov models can be used. According to one embodiment of the present invention, the language model 108 is a context-free grammar, such as a finite state grammar, that is a compact way of representing an exhaustive list of each and every word that the speech recognition

15     system 100 can recognize. Alternate embodiments of system 100 can use different types of language models, including a conventional $n$-gram statistical language model (such as a bigram model, where $n = 2$ ), where the probability of every word depends only on the $n - 1$ previous words. Hidden Markov models, $n$-gram language models, and context-free grammars are well-known to those skilled in the art and thus will not be discussed

20     further except as they pertain to the present invention.

An adapted LSA space derivation unit 111 receives the sequence of words and documents 110 and, using an LSA space 114, derives corresponding word and document vectors to adapt the LSA space 114 to reflect any changes in domain and/or style of the underlying language composition. The LSA space 114 is a continuous vector space

25     previously constructed from word and document vectors, referred to as semantic anchors 113, computed from a correlation matrix of word-document co-occurrences in a training

corpus of words and documents representative of a particular domain and style of language composition.

A semantic classification unit 112 receives the sequence of words and documents 110 and uses semantic inference to classify the sequence of words and documents 110 by determining the correlation between the sequence of words and documents 110 and one or more semantic anchors 113 present in the adapted LSA space 114. The correlation is the similarity between a vector corresponding to the sequence of words and documents 110 and the vectors corresponding to the semantic anchors 114 as determined by using a similarity measure. The semantic classification unit 112 classifies the sequence of words and documents 110 as corresponding to the semantic anchor 113 with the closest correlation. The semantic classification unit 112 sends a semantic representation 116 of the classified sequence of words and documents 110 to an application unit 118. The application unit 118 receives the semantic representation 116 and generates an application output 120.

The application unit 118 uses the semantic representation 116 to determine the application output 120 that is generated in response to the audio input 102. It should be noted that the speech recognition system 100 as described herein is used for illustrative purposes only, and there may be any number of other applications other than semantic inference for using the latent semantic adaptation of the present invention, including dictation, information retrieval, and word and document clustering applications. Such applications are well known to those skilled in the art, and thus will not be discussed further except as they pertain to the present invention.

FIG. 2 illustrates an overview of some of the components underlying LSA. As explained in more detail in FIG. 3, the basic LSA paradigm defines a mapping between a training text corpus $\mathcal{T}$ 202, i.e., a collection of $N$ documents of interest, $d_1, d_2, ... d_N$ 204, the underlying vocabulary $\mathcal{V}$ 206, $|\mathcal{V}| = M$, i.e., the set of all the words $w_1, w_2, ... w_M$ 208 appearing in the documents in $\mathcal{T}$, and a continuous vector space $\mathcal{S}$ 210, whereby each

word in $\mathcal{V}$ 206 and each document in $\mathcal{T}$ 202 is represented by a vector $\bar{d}_1...\bar{d}_N, \bar{w}_1...\bar{w}_M$ 212/214 in $\mathcal{S}$ 210. If $\mathcal{T}$ 202 is representative of general English, for example, then $\mathcal{V}$ 206 would be the M most frequent words in the language. Typically, M is on the order of a few thousand, and, depending on the application, N varies between a few hundred and several million documents; $\mathcal{T}$ 202 might comprise up to a billion words of text.

The continuous vector space $\mathcal{S}$ 210 is semantic in nature, because the "closeness" of vectors in the space $\mathcal{S}$ 210 is determined by the overall pattern of the language used in the training corpus $\mathcal{T}$ 202, as opposed to specific constructs. Hence, two words whose representations are "close" (in some suitable metric) tend to appear in the same kind of documents, whether or not they actually occur within identical word contexts in those documents. Conversely, two documents whose representations are "close" tend to convey the same semantic meaning, whether or not they contain the same word constructs. More generally, word and document vectors 212/214 associated with words and documents 204/208 that are semantically linked are also "close " in the space $\mathcal{S}$ 210. On the other hand, a semantic pattern not present in the training corpus $\mathcal{T}$ 202 cannot be inferred from the space $\mathcal{S}$ 210, hence the need to adapt the space to keep semantic knowledge as current as possible.

FIG. 3 illustrates selected components of the basic LSA paradigm 300 used to construct the continuous vector space $\mathcal{S}$, referenced in FIG. 3 as LSA space $\mathcal{S}$ 316. The LSA paradigm 300 first captures the semantic patterns of the word-document co-occurrences that appeared in the training corpus $\mathcal{T}$ 202 by constructing a word-document matrix $W$ 302 of dimension $M \times N$, whose entries $w_{ij}$ 304 suitably reflect the extent to which word $w_i$ 208 appeared in document $d_j$ 204, and then performing a singular value decomposition (SVD) of the word-document matrix $W$ 302 having an order of decomposition of $R << \min(M, N)$ as in [1]:

$$W = USV^T, \qquad (1)$$

where $U$ 306 is the $M \times R$ left singular matrix of row vectors, $u_i (1 \le i \le M)$, $S$ 308 is the $R \times R$ diagonal matrix of singular values $s_1 \ge s_2 \ge ...s_R > 0$, and $V^T$ is the transposition of $V$ 310, the $R \times N$ right singular matrix of row vectors $v_j (1 \le j \le N)$.

The value of $R$ can vary depending on the values of $M$ and $N$, and by balancing computational speed (associated with lower values of $R$) against accuracy (associated with higher values of $R$). Typical values for $R$ range from 5 to 100.

As is well-known to those skilled in the art, both left and right singular matrices $U$ 306 and $V$ 310 are column-orthonormal, i.e., $U^T U = V^T V = I_R$ (the identity matrix of order $R$). Thus, the column vectors of matrices $U$ 306 and $V$ 310 each define an orthonormal basis for the space of dimension $R$ spanned by the $u_i$'s and $v_j$'s. This is the LSA space $S$ 316, in which the scaled row vectors $\bar{u}_i = u_i S$ 318 and $\bar{v}_j = v_j S$ 320 (i.e. the rows of $US$ 312 and $VS$ 314) characterize the position of word $w_i$ and document $d_j$. For this reason, $\bar{u}_i$ 318 and $\bar{v}_j$ 320 are referred to as a word vector and a document vector, respectively.

Given the SVD from calculation (1), a particular document $d_j$ 204 in $W$ 202 can be determined based on the $j$th right singular vector $v_j$ according to the following calculation:

$$d_j = USv_j^T \qquad (2)$$

Further, based on calculation (2) and using well-known mathematical functions and properties, the value of the $j$th right singular vector $v_j$ can be calculated according to the following:

$$v_j = d_j^T US^{-1} \qquad (3)$$

It is to be appreciated that the value $US^{-1}$ does not change for different values of $j$, and therefore the value $US^{-1}$ can be pre-calculated and used during the classification of new words and documents 110 referenced in FIG. 1. This pre-calculation reduces the computation required to perform the functions of the semantic classification unit 112, thereby increasing the speed of a speech recognition system 100 during operation.

However, the LSA space $\mathcal{S}$ 316 generated by the basic LSA paradigm 300 is a fixed semantic space and must eventually be re-trained to keep pace with additions and changes not only to the domain, i.e. the underlying vocabulary $\mathcal{V}$ 206, but also to changes in the style of the documents. Otherwise, the semantic classification error rate of the

5    semantic classification unit 112 begins to increase as the new words and documents 110 vary from those contained in the original training corpus $\mathcal{T}$ 202. However, re-training by re-computing the LSA space $\mathcal{S}$ 316 generated by the basic LSA paradigm 300 is too computationally intensive to be of practical use.

FIG. 4 illustrates selected components of the adaptive LSA paradigm 400 using

10    latent semantic adaptation in accordance with an embodiment of the present invention. The adaptive LSA paradigm 400 extends the basic LSA paradigm 300 so that some or all of the data in new documents 110 are taken into account through incremental adaptation of the original LSA space $\mathcal{S}$ 316 in a way that is computationally efficient. Adaptation of the original LSA space $\mathcal{S}$ 316 insures that the semantic classification error rate of the

15    semantic classification unit 112 does not substantially increase as the new words and documents 110 vary from those contained in the original training corpus $\mathcal{T}$ 202.

The adaptive LSA paradigm 400 relies on two assumptions. The first assumption is that the dimension $R$ of the original LSA space $\mathcal{S}$ 316 is low enough that none of the corresponding $R$ singular values are zero. This is typically the case since the basic LSA

20    paradigm 300 seeks to operate at the maximum possible dimensionality reduction to increase computational speed without sacrificing accuracy. The second assumption is that the transformation necessary to adapt the original LSA space $\mathcal{S}$ 316 is invertible. If it was not, then a rather pathological situation would arise: the inability to go back to the original LSA space $\mathcal{S}$ 316 by simply forgetting the new data.

25    With reference to FIG. 4, if $n$ additional documents contain words drawn from the original underlying vocabulary $\mathcal{V}$ 206 plus $m$ words previously unseen (i.e. out-of-vocabulary words), then the adaptive LSA paradigm 400 constructs a word-document

matrix $\widetilde{W}$ 402 of dimension $(M+m)\times(N+n)$ in the same manner as described for generating matrix $W$ 202 in the basic LSA paradigm 300 in FIG. 3. Using the same order of decomposition $R$, the SVD of $\widetilde{W}$ 402 leads to:

$$\widetilde{W} = \widetilde{U}\widetilde{S}\widetilde{V}^T, \tag{4}$$

5    where $\widetilde{U}$ 406 is the left singular matrix of dimension $(M+m)\times R$, $\widetilde{S}$ 408 is the diagonal matrix of dimension $R\times R$, and $\widetilde{V}$ 410 is the right singular matrix of dimension $(N+n)\times R$, each having the same definitions and properties as described above for $W$, $U$, $S$, and $V$ in FIG. 3.

As shown in FIG. 4, the $m$ new words are gathered in the $m\times(N+n)$ matrix

10    $\widetilde{C} = [CE]$ 422, the $n$ new documents are gathered in the $(M+m)\times n$ matrix $\widetilde{D} = [D^T E^T]^T$ 424. $\widetilde{U}$ 406 is expressed as $[\widetilde{U}_1^T \widetilde{U}_2^T]^T$, where $\widetilde{U}_1^T$ 436 is the transposition of the left singular matrix of dimension $M\times R$ and $\widetilde{U}_2^T$ 438 is the transposition of the left singular matrix of dimension $m\times R$. $\widetilde{V}^T$ 410 is expressed as $[\widetilde{V}_1^T \widetilde{V}_2^T]$ where $\widetilde{V}_1^T$ 439 is the transposition of the right singular matrix of dimension $R\times N$ and $\widetilde{V}_2^T$ 440 is the

15    transposition of the right singular matrix of dimension $R\times n$. The new decomposition of $\widetilde{W}$ expressed in (4) leads to a different LSA space $\widetilde{S}$ 416, in which the word and document vectors are now given by the scaled row vectors $\overline{\widetilde{u}}_i = \widetilde{u}_i\widetilde{S}$ 418 and $\overline{\widetilde{v}}_j = \widetilde{v}_j\widetilde{S}$ 420 (i.e. the rows of $\widetilde{U}\widetilde{S}$ 412 and $\widetilde{V}\widetilde{S}$ 414) to characterize the position of word $w_i$ and document $d_j$.

20    FIG. 7 illustrates the prior art approach referred to as baseline adaptation, where the distinction between the SVD in (1) of the original word-document co-occurrence matrix $W$ 302 in FIG. 3 and the SVD in (4) of the extended word-document co-occurrence matrix $\widetilde{W}$ 402 in FIG. 4 is ignored by making the (obviously invalid) assumption that the original LSA space $S$ 316 is the same as the new LSA space $\widetilde{S}$ 416.

25    In other words, in baseline adaptation, the SVD in (1) is still assumed to be valid even after the new documents become available, and the problem is reduced to representing the new data in the original LSA space $S$ 316.

Referring now to FIGS. 4 and 7, the baseline adaptation approach treats the portions of the matrix $\widetilde{W}$ 402 identified as $C$ 430 and $D$ 432 as merely extensions of additional rows or columns of the original matrix $W$ 302, and discards altogether the portion of the extended matrix $\widetilde{W}$ 402 identified as $E$ 434. This has the effect of ignoring significant amounts of new data, including any out-of-vocabulary words in the new documents.

Using the baseline adaptation approach, the representation of those portions of the new data that will be added to the original LSA space $S$ 316 is obtained from the SVD of as $C$ 430 and $D$ 432 as follows:

$$C = YSV^T, \tag{5}$$

$$D = USZ^T, \tag{6}$$

where the $m \times R$ matrix $Y$ 426 and the $n \times R$ matrix $Z$ 428 are defined *a posteriori* (as plug-ins), to satisfy the relationship. In essence, using the baseline adaptation framework 700, the role of matrices $Y$ 426 and $Z$ 428 is to "extend" the original matrices $U$ 306 and $V$ 310 to accommodate the new data. The original word and document vectors $\overline{u}_i$ 318 and $\overline{v}_j$ 320 are still given by the rows of $US$ 312 and $VS$ 314, but the new word and document vectors $\overline{y}_i$ 446 and $\overline{z}_j$ 448 are given by the rows of $YS$ 442 and $ZS$ 444, respectively. From (5) and (6), these are seen to be:

$$YS = CV, \tag{7}$$

$$ZS = D^TU. \tag{8}$$

The effect, illustrated in FIG. 7, is that the original LSA space $S$ 316 becomes populated with the new data, i.e. the new word and document vectors $\overline{y}_i$ 446 and $\overline{z}_j$ 448, hence the name "folding-in."

A major drawback to the above-described baseline adaptation approach illustrated in FIG. 7 is poor performance, since even when populated with the new word and document vectors $\overline{y}_i$ 446 and $\overline{z}_j$ 448, the misclassification error rate using the original LSA space $S$ 316 is still high when the new words and documents vary from the original

training corpus $\widetilde{7}$ 202, e.g. when the new documents contain several new words not in the original training corpus.

In contrast, the latent semantic adaptation approach of the present invention achieves significant reductions in the misclassification error rate. Unlike baseline adaptation, the latent semantic adaptation approach of the present invention recognizes that there is an important distinction between the SVD in (1) of the original word-document co-occurrence matrix $W$ 302 in FIG. 3 and the SVD in (4) of the extended word-document co-occurrence matrix $\widetilde{W}$ 402 in FIG. 4 that must be taken into account since the original LSA space $S$ 316 is <u>not</u> the same as the new LSA space $\widetilde{S}$ 416. In other words, the SVD in (1) is no longer valid after the new documents become available, so the problem is more than just representing the new data in the original LSA space $S$ 316. Therefore, in one embodiment, the latent semantic adaptation approach treats the portions of the matrix $\widetilde{W}$ 402 identified as $C$ 430 and/or $D$ 432 in FIG. 4 as new data that must be accounted for in a new LSA space $\widetilde{S}$ 416. In one embodiment, the portion of the matrix $\widetilde{W}$ 402 identified as $E$ 434 in FIG. 4 is also treated as new data that must be accounted for in a new LSA space $\widetilde{S}$ 416.

In one embodiment of latent semantic adaptation, the scaled row vectors (i.e. the rows of $\widetilde{U}\widetilde{S}$ 412 and $\widetilde{V}\widetilde{S}$ 414) are obtained directly from the SVD of the entire matrix $\widetilde{W}$ 402 in (4) using a latent semantic adaptation framework 400 as defined in the equations that follow. By inspection from FIG. 4,

$$C = \widetilde{U}_2 \widetilde{S} \widetilde{V}_1^T, \tag{9}$$

$$D = \widetilde{U}_1 \widetilde{S} \widetilde{V}_2^T, \tag{10}$$

and

$$W = \widetilde{U}_1 \widetilde{S} \widetilde{V}_1^T, \tag{11}$$

$$E = \widetilde{U}_2 \widetilde{S} \widetilde{V}_2^T, \tag{12}$$

each of which are column-orthonormal, i.e., $\widetilde{U}^T\widetilde{U} = \widetilde{V}^T\widetilde{V} = I_R$ (the identity matrix of order $R$). The orthogonality constraints can also be expressed in terms of $\widetilde{U}_1$, $\widetilde{U}_2$, $\widetilde{V}_1$, and $\widetilde{V}_2$ as follows:

$$\widetilde{U}^T\widetilde{U} = I_R = \widetilde{U}_1^T\widetilde{U}_1 + \widetilde{U}_2^T\widetilde{U}_2, \tag{13}$$

$$\widetilde{V}^T\widetilde{V} = I_R = \widetilde{V}_1^T\widetilde{V}_1 + \widetilde{V}_2^T\widetilde{V}_2. \tag{14}$$

In one embodiment, the foregoing equations (9)-(14) define the latent semantic adaptation framework 400 of the method of the present invention. The latent semantic adaptation framework 400 is used to solve for the "extension" SVD matrices $\widetilde{U}$ 406, $\widetilde{S}$ 408, and $\widetilde{V}$ 410 as a function of the original SVD matrices $U$ 306, $S$ 308, $V$ 310, and "extension" SVD matrices $Y$ 426, and $Z$ 428.

According to one embodiment, the solution is obtained by setting up a latent semantic adaptation transformation 500, as illustrated in FIG. 5, based on the assumptions previously noted that the dimension $R$ of the original LSA space $S$ 316 is low enough that none of the corresponding $R$ singular values are zero, and that the transformation necessary to adapt the original LSA space $S$ 316 is invertible. Starting with $\widetilde{S}$ 408, the shift from $S$ 308 in FIG. 3 to $\widetilde{S}$ 408 in FIG. 4 can be captured as illustrated in FIG. 5 by the following expressions:

$$\widetilde{U}_1 = U\widetilde{G}, \tag{15}$$

$$\widetilde{V}_1 = V\widetilde{H}, \tag{16}$$

where $\widetilde{G}$ 508 and $\widetilde{H}$ 518 are $(R \times R)$ matrices that, according to the second assumption, are assumed to be invertible. Taken together, (15) and (16) define a latent semantic adaptation matrix transformation 500 to apply to the original SVD matrices $U$ 306 and $V$ 310 to update them according to the new data.

It is fairly straightforward to show that matrix transformation 500 also applies to the "extension" SVD matrices $Y$ 426, and $Z$ 428 resulting from the "folding-in" process, designated in FIG. 4 as $\widetilde{U}_2$ 438 and $\widetilde{V}_2$ 440, respectively. Specifically, first note that (1) and (11), together with (15) and (16) and the orthogonality properties of $U$ and $V$, lead to:

$$S = \widetilde{G}\widetilde{S}\widetilde{H}^T. \tag{17}$$

Since matrices $\widetilde{G}$ 508 and $\widetilde{H}$ 518 are invertible and, in accordance with the first

assumption, since both $S$ 308 and $\widetilde{S}$ 408 are assumed to contain no zero singular value,

implies the following equivalent identities:

$$\widetilde{G} = S\widetilde{H}^{-T}\widetilde{S}^{-1}, \tag{18}$$

$$\widetilde{H} = S\widetilde{G}^{-T}\widetilde{S}^{-1}, \tag{19}$$

where the latter identity exploits the fact that $S$ 308 and $\widetilde{S}$ 408 are diagonal.

On the other hand, equations (5) and (9), together with (16), can be expressed as:

$$YS = \widetilde{U}_2\widetilde{S}\widetilde{H}^T, \tag{20}$$

while equations (6) and (10), together with (15) yield:

$$SZ^T = \widetilde{G}\widetilde{S}\widetilde{V}_2^T. \tag{21}$$

Thus, exploiting equation (18) in (20) and equation (19) in (21), $\widetilde{U}_2$ 438 and $\widetilde{V}_2$ 440 can be

obtained, after re-arranging, as:

$$\widetilde{U}_2 = Y\widetilde{G}, \tag{22}$$

$$\widetilde{V}_2 = Z\widetilde{H}. \tag{23}$$

Equations (22)-(23) provide a convenient way to express the "extension" SVD matrices

$\widetilde{U}$ 406 and $\widetilde{V}$ 410 in terms of the matrices $\widetilde{G}$ 508 and $\widetilde{H}$ 518 of the latent semantic

adaptation matrix transformation 500 postulated in expressions (15)-(16), as follows:

$$\widetilde{U} = \begin{bmatrix} U \\ Y \end{bmatrix}\widetilde{G}, \tag{24}$$

$$\widetilde{V} = \begin{bmatrix} V \\ Z \end{bmatrix}\widetilde{H}. \tag{25}$$

The issue is now to solve for matrices $\widetilde{G}$ 508 and $\widetilde{H}$ 518 as a function of the original

entities.

The first step in solving for matrices $\widetilde{G}$ 508 and $\widetilde{H}$ 518 is to take advantage of

equations (15)-(16) and (24)-(25) in (13)-(14). The orthogonality properties of $U$ 306

and $V$ 310 are used to obtain:

$$\widetilde{G}^T(I_R + Y^TY)\widetilde{G} = I_R, \tag{26}$$

$$\widetilde{H}^T(I_R + Z^TZ)\widetilde{H} = I_R. \tag{27}$$

Again invoking the assumption regarding the non-singularity of matrices $\widetilde{G}$ 508

and $\widetilde{H}$ 518, equations (26)-(27) may be re-written as:

$$\widetilde{G}^{-T}\widetilde{G}^{-1} = (\widetilde{G}\widetilde{G}^T)^{-1} = (I_R + Y^TY), \tag{28}$$

$$(\widetilde{H}^{-T}\widetilde{H}^{-1} = (\widetilde{H}\widetilde{H}^T)^{-1} = (I_R + Z^TZ), \tag{29}$$

or, equivalently:

$$(\widetilde{G}\widetilde{G}^T) = (I_R = Y^TY)^{-1}, \tag{30}$$

$$(\widetilde{H}\widetilde{H}^T) = (I_R + Z^TZ)^{-1} \tag{31}$$

Equations (30)-(31) can in turn be used to derive $\widetilde{G}$ 508 as a function of "extension" SVD

matrix $Y$ 426, and $\widetilde{H}$ 518 as a function of "extension" SVD matrix $Z$ 428. Note that the

inverse appearing in the right hand side of equations (30) and (31) may not have to be

computed directly. Recall the well-known matrix identity:

$$(A + P^TQ)^{-1} = A^{-1} - (A^{-1}P^T)(I + QA^{-1}P^T)^{-1}(QA^{-1}), \tag{32}$$

for any nonsingular (square) matrix $A$ and matrices $P$, $Q$ with compatible dimensions.

Applied to (30)-(31), this results in:

$$(\widetilde{G}\widetilde{G}^T) = I_R - Y^T(I_m + YY^T)^{-1}Y, \tag{33}$$

$$(\widetilde{H}\widetilde{H}^T) = I_R - Z(I_n + ZZ^T)^{-1}Z, \tag{34}$$

which may be computationally beneficial if $m < R$ and/or $n < R$. Regardless of how the

inverse is computed, once the right hand side is known, the computation of matrices $\widetilde{G}$

508 and $\widetilde{H}$ 518 can be done efficiently through Choleski decomposition, or, in the

symmetric case, through matrix square root computation.

To be able to derive the new vectors $\widetilde{US}$ 412 and $\widetilde{VS}$ 414 that are to populate the new LSA space $\widetilde{S}$ 416, the matrix $\widetilde{S}$ 408 must be expressed as a function of known quantities. But observe from equations (24)-(25) that:

$$\widetilde{US} = \begin{bmatrix} U \\ Y \end{bmatrix} \widetilde{GS}, \tag{35}$$

$$\widetilde{VS} = \begin{bmatrix} V \\ Z \end{bmatrix} \widetilde{HS}. \tag{36}$$

Thus, it is sufficient to find suitable expressions for $\widetilde{GS}$ and $\widetilde{HS}$.

From equations (17), (28), and (29), it is clear that:

$$(\widetilde{GS})(\widetilde{GS})^T = \widetilde{GS}^2 \widetilde{G}^T = S\widetilde{H}^{-T}\widetilde{H}^{-1}S = S(I_R + Z^T Z)S, \tag{37}$$

$$(\widetilde{HS})(\widetilde{HS})^T = \widetilde{HS}^2 \widetilde{H}^T = S\widetilde{G}^{-T}\widetilde{G}^{-1}S = S(I_R + Y^T Y)S. \tag{38}$$

Thus, it is also possible to obtain $\widetilde{GS}$ and $\widetilde{HS}$ directly through Choleski decomposition, in a manner analogous to that mentioned above $\widetilde{G}$ 508 and $\widetilde{H}$ 518. In fact, as illustrated in FIG. 6, if $J$ 608 and $K$ 618 are the solutions of relevant Choleski decompositions, viz.:

$$JJ^T = (I_R + Y^T Y), \tag{39}$$

$$KK^T = (I_R + Z^T Z), \tag{40}$$

then equations (35)-(38) admit as solutions:

$$\widetilde{US} = \begin{bmatrix} US \\ YS \end{bmatrix} K, \tag{41}$$

$$\widetilde{VS} = \begin{bmatrix} VS \\ ZS \end{bmatrix} J. \tag{42}$$

In other words, in accordance with one embodiment of the present invention, the original vectors $US$ 312 and $VS$ 314, as well as the new vectors resulting from the "folding-in" process $YS$ 442 and $ZS$ 444, can be transformed using a latent semantic adaptation vector transformation 600 defined by the transformation matrices $K$ 618 and $J$ 608 to respectively yield the updated word vectors $\widetilde{US}$ 412 and document vectors $\widetilde{VS}$

414. Therefore, equations (41) and (42) make it possible to adapt the original LSA space $S$ 316 of FIG. 3 to the new LSA space $\tilde{S}$ 416 of FIG. 4.

In one embodiment of the latent semantic adaptation framework 400, the new information, as reflected through the transformation matrices $K$ 618 and $J$ 608, affects both original word and document vectors $\bar{u}_i$ 318 and $\bar{v}_j$ 320 and new word and document vectors $\bar{y}_i$ 446 and $\bar{z}_j$ 448, referred to as two-sided adaptation. Stated another way, the transformed representation of the new word and document vectors $\bar{y}_i$ 446 and $\bar{z}_j$ 448 takes into account its own influence on the underlying semantic knowledge that was encapsulated in the original LSA space $S$ 316 of FIG. 3 (i.e. the existing word and document vectors $\bar{u}_i$ 318 and $\bar{v}_j$ 320) to yield the transformed word and document vectors $\tilde{\bar{u}}_i$ 418 and $\tilde{\bar{v}}_j$ 420 that populate the new LSA space $\tilde{S}$ 416 of FIG. 4. As indicated by the arrows in the new LSA space $\tilde{S}$ 416 of FIG. 4, the positions of both the words and documents represented by original word and document vectors $\bar{u}_i$ 318 and $\bar{v}_j$ 320 have shifted from their positions in the original LSA space $S$ 316 to reflect their changed position (i.e. their relationship) within the new LSA space $\tilde{S}$ 416. The new LSA space $\tilde{S}$ 416 allows not only for improvements in the misclassification error rate, but also provides the ability to adapt the speech recognition database that embodies the new LSA space $\tilde{S}$ 416 in real-time, because the application of the transformation matrices $K$ 618 and $J$ 608 is computationally efficient and bypasses the need to re-compute the LSA space.

In another embodiment of the latent semantic adaptation framework 400 of the present invention, only the new documents are considered and not the new words appearing in those new documents, referred to as one-sided adaptation. As in two-sided adaptation, one-sided adaptation does not simply "fold-in" the new documents, but rather generates a transformed representation of the new document vectors $\tilde{\bar{v}}_j$ 420 to generate a new LSA space $\tilde{S}$ 416. While not as dramatic an improvement of the misclassification rate is obtained with one-sided adaptation as can be obtained with two-sided adaptation,

in certain applications of a speech recognition system 100, one-sided adaptation may be sufficient to allow for real-time adaptation of the original LSA space $S$ 316.

In addition to providing improved performance through lowering the misclassification rate, it is also worth noting that the latent semantic adaptation framework 400 and resulting latent semantic adaptation matrix and vector transformations 500 and 600 respectively are computationally efficient. Compared to the "folding-in" computations of the baseline adaptation approach, the latent semantic adaptation matrix and vector transformations 500 and 600 of the latent semantic adaptation framework 400 entail less overhead. For example, in terms of the number of floating point operations required, the overhead associated with the latent semantic adaptation vector transformations 600 embodied in equations (39)-(42) can be expressed as:

$$N_{adapt} = \frac{2}{3}R^3 + [(M+N) + 2(m+n) - 1]R^2 + (m+n+1)R. \qquad (43)$$

For typical values of the various dimensions involved, expression (43) will be dominated by $(M+N)R^2$. Depending on the application, this quantity may fall anywhere between about 50 million (for voice command and control types of speech recognition applications using a limited vocabulary) and more than 1 billion (for large vocabulary transcription). Still, on current high-end machines, this quantity only represents up to a few seconds of central processor unit (CPU) time. Compared to recomputing the SVD from scratch, which requires $O(MNR)$ operations, the computational complexity is reduced by a factor of approximately $\min(M, N) / R$. In many speech recognition applications, the reduction factor will be on the order of 1000. In such cases, the latent semantic adaptation framework 400 and resulting latent semantic adaptation matrix and vector transformations 500 and 600 make it practical to adapt the new LSA space $\tilde{S}$ 416 with real-time word and document updates, whereas SVD re-computation would generally not be feasible.

In some embodiments of the latent semantic adaptation framework 400 of the present invention it may be better to proceed separately in the estimation of the scaled row vectors $\widetilde{US}$ 412 and $\widetilde{VS}$ 414 (or possibly, the transformation matrices $\widetilde{G}$ 508 and $\widetilde{H}$ 518) that comprise the transformed word and document vectors $\bar{\bar{u}}_i$ 418 and $\bar{\bar{v}}_j$ 420 respectively, while in other embodiments it may be better to first compute one and then use the results to compute the other. The choice depends on what piece of evidence can be considered the most reliable.

For example, in speech recognition applications for dictation it is likely that the new data would be primarily new documents, with a few occasional new words. This is because the vocabulary there is very large (e.g., 60,000 words), so most of the words in a new document (e.g., letter) would already be known. In speech recognition applications for command and control, on the other hand, the vocabulary is more limited, so it is likely that the new data would contain new words. In the context of semantic inference, for example, the same command can be expressed in alternative ways, using different words, e.g. "make a new spreadsheet" when the pre-defined wording of the command is "Open Microsoft Excel."

In one embodiment, if the new data primarily involves adding new documents, then both matrix $D$ 432 and matrix $E$ 434 would be reliable starting points since they contain the new document and new word data for the new documents, while matrix $C$ 430 might not, since it contains only the new word data. In that case, it follows that "extension" SVD matrix $Z$ 428 (computed from equation (8)) is more reliable than "extension" SVD matrix $Y$ 426 (computed from equation (7)), which means that transformation matrix $K$ 618 is more reliable than transformation matrix $J$ 608 . Consequently, it is better to first compute the transformed word vectors $\bar{\bar{u}}_i$ 418 from $\widetilde{US}$ 412 using the transformation matrix $K$ 618 in equation (41). Instead of computing "extension" SVD matrix $Y$ 426 from the less reliable matrix $C$ 430 in equation (7), "extension" SVD matrix $Y$ 426 can now be obtained from equation (44) as:

$$E = YSZ^T, \tag{44}$$

which, in turn, implies:

$$Y = EZ(Z^TZ)^{-1}S^{-1}. \tag{45}$$

Thus, if the new data primarily involves adding new documents, then "extension" SVD

matrix $Y$ 426, i.e. the new word data, can be secondarily obtained through (45) for use in

(39) and then (42), which completes the transformation.

On the other hand, in an alternate embodiment, if the new data primarily involves

adding new words, then both matrices $C$ 430 and $E$ 434 would be reliable starting points

since they contain all of the new word data for the new documents, while matrix $D$ 432

might not, since it contains only the new document data. In that case, it follows that

"extension" SVD matrix $Y$ 426 (computed from equation (7)) is more reliable than

"extension" SVD matrix $Z$ 428 (computed from equation (8)), which means that

transformation matrix $J$ 608 is more reliable than transformation matrix $K$ 618. In that

case it is advisable to first compute the transformed document vectors $\bar{v}_j$ 420 from $\widetilde{VS}$

414 using transformation matrix $J$ 608 in equation (42). Instead of computing

"extension" SVD matrix $Z$ 428 from the less reliable matrix $D$ 432 in equation (8),

"extension" SVD matrix $Z$ 428 can now be obtained from equation (44) as:

$$Z = E^TY(Y^TY)^{-1}S^{-1}. \tag{46}$$

Equation (46) can, in turn, be used in (40) and then (41), which completes the

transformation.

From the foregoing, it is apparent that equations (45) and (46) provide a

convenient way to compute "extension" SVD matrix $Y$ 426 as a function of "extension"

SVD matrix $Z$ 428, or vice versa, as necessary. It should be noted that other alterations

in the order of computation of the equations of the matrix and vector transformations 500

and 600 resulting from the latent semantic adaptation framework 400 might be used to

accommodate the different word and document usage characteristics of different speech

recognition systems and applications without departing from the scope of the invention.

Turning now to FIG. 8, the particular methods of the invention are described in terms of computer software with reference to a flowchart. The methods to be performed by a computer constitute computer programs made up of computer-executable instructions. Describing the methods by reference to a flowchart enables one skilled in the art to develop such programs including such instructions to carry out the methods on suitably configured computers (the processor of the computer executing the instructions from computer-accessible media). The computer-executable instructions may be written in a computer programming language or may be embodied in firmware logic. If written in a programming language conforming to a recognized standard, such instructions can be executed on a variety of hardware platforms and for interface to a variety of operating systems. In addition, the present invention is not described with reference to any particular programming language. It will be appreciated that a variety of programming languages may be used to implement the teachings of the invention as described herein. Furthermore, it is common in the art to speak of software, in one form or another (e.g., program, procedure, process, application...), as taking an action or causing a result. Such expressions are merely a shorthand way of saying that execution of the software by a computer causes the processor of the computer to perform an action or a produce a result.

FIG. 8 is a flowchart illustrating the process followed in using latent semantic adaptation to generate a new LSA space $\tilde{S}$ 416 according to one embodiment of the present invention. Initially, at process block 810, a speech recognition system 100 such as the one illustrated in FIG. 1, generates an original LSA space $S$ 316 from a singular value decomposition (SVD) of a word-document correlation matrix $W$ 302 of word-document co-occurrences in a training text corpus $\mathcal{T}$ 202 of words and documents representative of a particular domain and style of language composition. The SVD results in original left and right singular matrices $U$ 306 and $V$ 310, respectively, and diagonal matrix $S$ 308, where the rows of $US$ 312 and $VS$ 314 characterize the position

of the words and documents of training text corpus $\mathcal{T}$ 202 in the original LSA space $\mathcal{S}$ 316.

In one embodiment, after the training process 810 is completed, an adapted LSA space derivation unit 111 employed in the speech recognition system 100 continues at process block 820 to gather new words and documents in the extension matrices $C$ 430, $D$ 432 and $E$ 434, which extend the data in the original word-document correlation matrix $W$ 302 (i.e. the new words and documents not present in the original training text corpus $\mathcal{T}$ 202). Using an SVD, the adapted LSA space derivation unit 111, obtains the "extension" SVD matrices $Y$ 426 and $Z$ 428 which extend the original left and right singular matrices $U$ 306 and $V$ 310 by "folding-in" the new words and new documents, respectively.

In one embodiment, processing continues at process block 830, where the adapted LSA space derivation unit 111 may optionally apply the transformation matrices $\widetilde{G}$ 508 and $\widetilde{H}$ 518 to the left and right singular matrices $U$ 306 and $V$ 310, to update them with the new words and new documents, respectively. In addition, in one embodiment, the adapted LSA space derivation unit 111 may optionally apply transformation matrices $\widetilde{G}$ 508 and $\widetilde{H}$ 518 to the "extension" SVD matrices $Y$ 426 and $Z$ 428, to update them with the new words and documents, respectively. In normal operation, however, process block 830 is unnecessary since only the updated word and document vectors $\widetilde{US}$ 412 and $\widetilde{VS}$ 414 are actually needed to generate the new LSA space $\widetilde{S}$ 416 (as described in process block 840, below).

In one embodiment, processing continues at process block 840, where the adapted LSA space derivation unit 111 derives the updated word and document vectors $\widetilde{US}$ 412 and $\widetilde{VS}$ 414 by applying the transformation matrices $J$ 608 and $K$ 618 to the word and document vectors $US$ 312 and $VS$ 314, as well as the extended word and document vectors $YS$ 442 and $ZS$ 444, respectively.

In general, a significant improvement in the misclassification error rate of the speech recognition system 100 is obtained by simultaneously updating both the word and document vectors. However, in some embodiments of latent semantic adaptation, the improvement may be even more significant as well as more efficient if, at processing

5    blocks 820 - 840, the adapted LSA space derivation unit 111 first derives the updated word vectors $\widetilde{US}$ 412, and then derives the document vectors $\widetilde{VS}$ 414, or vice versa. This is because when the new data primarily involves adding new documents, for example, the "extension" SVD matrix $Y$ 426 can be obtained from the values already computed for $E$ 434, $Z$ 428, and $S$ 308 using equation (45), whereas when the new data primarily

10   involves adding new words, the "extension" SVD matrix $Z$ 428, can be obtained from the values already computed for $E$ 434, $Y$ 426, and $S$ 308 using equation (46). In other embodiments, it may be sufficient to process only the new documents using a one-sided adaptation approach instead of the full two-sided adaptation.

In one embodiment, processing continues at process block 850, where the adapted

15   LSA space derivation unit 111 generates the new LSA space $\widetilde{S}$ 416 by populating it with the newly derived updated word and document vectors $\widetilde{US}$ 412 and $\widetilde{VS}$ 414 obtained at process block 840, i.e. the scaled row vectors $\overline{\overline{u}}_i$ 418 and $\overline{\overline{v}}_j$ 420.

FIG. 9 illustrates one embodiment of a computing device suitable for use with one embodiment the present invention. As illustrated, the speech recognition system 100 of

20   FIG. 1 may be implemented on a computer system 900. Computer system 900 includes processor 902, display device 906, and input/output (I/O) devices 908, coupled to each other via a bus 910. Additionally, a memory subsystem 912, which can include one or more of cache memories, system memory (RAM), and nonvolatile storage devices (e.g., magnetic or optical disks) is also coupled to bus 910 for storage of instructions and data

25   for use by processor 902. I/O devices 908 represent a broad range of input and output devices, including keyboards, cursor control devices (e.g., a trackpad or mouse), microphones to capture the voice data, speakers, network or telephone communication

interfaces, printers, etc. Computer system 900 also includes well-known audio

processing hardware and/or software to transform analog voice data to a digital form

which can be processed by the speech recognition system 100 implemented in computer

system 900. In addition to personal computers, laptop computers, and workstations, in

5       some embodiments, computer system 900 may be incorporated in a mobile computing

device such as a personal digital assistant (PDA) or mobile telephone without departing

from the scope of the invention.

Components 902 - 912 of computer system 900 perform their conventional

functions known in the art. Collectively, these components are intended to represent a

10      broad category of hardware systems, including but not limited to general purpose

computer systems based on the PowerPC® processor family of processors available from

Motorola, Inc. of Schaumburg, Illinois, or the Pentium® processor family of processors

available from Intel Corporation of Santa Clara, California.

It is to be appreciated that various components of computer system 900 may be re-

15      arranged, and that certain implementations of the present invention may not require nor

include all of the above components. For example, a display device may not be included

in system 900. Additionally, multiple buses (e.g., a standard I/O bus and a high

performance I/O bus) may be included in system 900. Furthermore, additional

components may be included in system 900, such as additional processors (e.g., a digital

20      signal processor), storage devices, memories, network/communication interfaces, etc.

In the illustrated embodiment of FIG. 9, the method and apparatus for speech

recognition using latent semantic adaptation with word and document updates according

to the present invention as discussed above is implemented as a series of software

routines run by computer system 900 of FIG. 9. These software routines comprise a

25      plurality or series of instructions to be executed by a processing system in a hardware

system, such as processor 902 of FIG. 9. Initially, the series of instructions are stored on

a storage device of memory subsystem 912. It is to be appreciated that the series of

instructions can be stored using any conventional computer-readable or machine-accessible storage medium, such as a diskette, CD-ROM, magnetic tape, DVD, ROM, Flash memory, etc. It is also to be appreciated that the series of instructions need not be stored locally, and could be stored on a propagated data signal received from a remote

5    storage device, such as a server on a network, via a network/communication interface. The instructions are copied from the storage device, such as mass storage, or from the propagated data signal into a memory subsystem 912 and then accessed and executed by processor 902. In one implementation, these software routines are written in the C++ programming language. It is to be appreciated, however, that these routines may be

10   implemented in any of a wide variety of programming languages.

These software routines are illustrated in memory subsystem 912 as speech recognition instructions 920, latent semantic adaptation instructions 922, latent semantic classification instructions 924, and action generation instructions 923. Also illustrated are analog to digital (A/D) transformation instructions 925, acoustic model(s) 926, and

15   language model(s) 927 that support the speech recognition system 100.

In the illustrated embodiment, the memory subsystem 912 of FIG. 9 also includes the semantic anchors 928 that comprise the word and document vectors of the LSA spaces 312 and 412. In one embodiment, the semantic anchors 928 are implemented in a speech recognition database using any of a wide variety of database formats known in the

20   art. As with the software instructions, the semantic anchors 928 may be copied from a storage device, such as mass storage, or from a propagated data signal into the memory subsystem 912 and then accessed and executed by processor 902.

In alternate embodiments, the present invention is implemented in discrete hardware or firmware. For example, one or more application specific integrated circuits

25   (ASICs) could be programmed with the above described functions of the present invention. By way of another example, speech recognition unit 104, adapted LSA space derivation unit 111, semantic classification unit 112, and application unit 118 of FIG. 1

could be implemented in one or more ASICs of an additional circuit board for insertion into hardware system 900 of FIG 9.

In the discussions above, the present invention is described with reference to speech recognition systems. It is to be appreciated, however, that alternate embodiments of the present invention can be used with other types of pattern recognition systems, such as visual rather than audio pattern recognition, handwriting recognition systems (e.g., optical character recognition (OCR)), etc.

It is to be appreciated that the method and apparatus for speech recognition using latent semantic adaptation with word and document update of the present invention can be employed in any of a wide variety of manners. By way of example, a speech recognition system employing latent semantic adaptation with word and document update could be used in conventional personal computers, security systems, home entertainment or automation systems, etc.

The performance of the above system was tested in the context of the "Speakable Items" desktop command and control task defined on the MacOS operating system. FIG. 10 illustrates a graph 1000 showing the misclassification error rates 1002 of five different approaches to command classification versus the number of variants 1004 of a collection of canonical commands from a number of different speakers. During performance testing, the five different approaches that were considered were: (i) full re-computation of the LSA space from scratch 1006, which served as a benchmark against which to measure the other setups; (ii) constant dimension re-computation of the LSA space 1008, where the LSA space is re-computed from scratch, but the LSA dimension remains constant at R = 100; (iii) baseline adaptation of the LSA space 1010 using traditional "folding-in" of the new words and documents in the existing LSA space; (iv) one-sided adaptation of the LSA space 1012, where only the new documents are used to adapt the LSA space using one embodiment of latent semantic adaptation in accordance with the present invention; and (v) two-sided adaptation of the LSA space 1014, where both the new documents as

well as the new words contained in the new documents are used to adapt the LSA space using one embodiment of latent semantic adaptation in accordance with the present invention.

As shown in FIG. 10, the two-sided adaptation approach 1014 remains competitive with constant dimension re-computation 1008 when the new words and documents vary from the original training corpus. In other words, the two-sided adaptation approach 1014 achieves a lower misclassification error rate 1002 than the baseline adaptation approach 1010 as the number of variant instances 1004 increases. Although less optimal than the two-sided approach, the same holds true for the one-sided adaptation approach 1012. Moreover, the competitive performance of the one-sided and two-sided latent semantic adaptation approaches 1012 and 1014 is achieved at a computation cost only slightly greater than the traditional "folding-in" of the baseline adaptation approach 1010, which has a much lower level of performance.

Therefore, a method and apparatus for speech recognition using latent semantic adaptation has been described. An audio input is provided to a speech recognizer that identifies the words in the input. The sequence of words comprising the audio input are then provided to an adapted LSA space generation unit that first trains the speech recognition system by generating an LSA space that reflects the semantic knowledge represented by a training corpus of words and documents, and then continually adapts the LSA space to reflect the semantic knowledge represented by the new words and documents as they become available. The resulting adapted LSA space can then provided to an application such as a semantic classifier that classifies the audio input as corresponding to a particular command, word, or sequence of words, depending on the application. This classification process is advantageously based on an adapted semantic representation of all of the words and documents that comprise the audio input rather than on a semantic representation of just the words and documents in the original training corpus. Thus, any application that needs to quickly and accurately recognize a particular

speech utterance associated with a semantic representation may employ the adapted

semantic representation provided by the method and apparatus of the present invention.

The adapted semantic representation of all of the words and documents comprising the

audio input advantageously allows the present invention to accurately recognize speech in

5    real-time, even when the speech includes words and documents not in the original

training corpus. Finally, whereas many alterations and modifications of the present

invention will be comprehended by a person skilled in the art after having read the

foregoing description, it is to be understood that the particular embodiments shown and

described by way of illustration are in no way intended to be considered limiting.

10    References to details of particular embodiments are not intended to limit the scope of the

claims.